

1. I would start by creating a Test Plan for the product. For this I will gather and analyze all requirements, identify user stories and their respective test cases. I would go through API documentation i.e. endpoints, parameters, expected response, etc. This will give both QA and Dev team an insight to all aspects for the product to make a complete test plan and strategy.
2. For each new functionality I would refer to the user stories and test cases for each functionality identified in the previous part. Based on those TCs, QA team will create test environment/data which will be utilized in test execution. If agile process is being followed then QA team will have all the requirements of the APIs under development so we can start our automation process early on. We can create an automation testing framework and depending on the automation cost we can automate the test cases of APIs as they reach the QA bucket. Later on all these automated TCs will be used in regression testing for the future sprints.
3. I would suggest following tools to QA team for daily tasks;
 - a. JIRA/Kanban board (for sprint, task and bug tracking)
 - b. Confluence (for test documentation)
 - c. TestRail (for maintaining test cases repository)
 - d. Postman (for API testing)
 - e. Slack (for communication)
4. For automation firstly, we will need to identify candidate test cases i.e. TCs which are executed repeatedly, TCs which require less effort to automate and their complexity. As mentioned in point 2 if we are following agile then we know about the requirement at the early stage so we can create/identify our test cases and start automating them early on. As we'll be doing API automation so I'll suggest following technology stack for automation;
 - a. Java
 - b. RestAssured
 - c. Cucumber
 - d. TestNG
 - e. IntelliJ IDEA (or any other IDE)

I will distribute TCs in 3 categories

- I. Most repeated and easy to implement
- II. Most repeated and complex to implement
- III. Less repeated

Once category 1 TCs are automated then we'll move to category 2 and after that category 3(if required and have sufficient resources).